

### **Amendments to the Claims:**

This listing of claims will replace all prior versions, and listings, of claims in the application:

### **Listing of Claims:**

1. (Currently amended) A method in a data processing system for processing instructions, the method comprising:  
responsive to receiving an instruction at a processor in the data processing system, determining whether an indicator is associated with the instruction; ~~[[and]]~~  
enabling counting of each event associated with execution of the instruction if the indicator is associated with the instruction; and  
counting each event associated with the execution of the instruction if counting is enabled for the instruction.
2. (Canceled)
3. (Currently amended) The method of claim ~~[[2,]]~~ 1, wherein the counting step comprises:  
incrementing a counter associated with the indicator each time the event occurs.
4. (Original) The method of claim 1, wherein the receiving step comprises:  
*receiving the instruction in an instruction cache in the processor.*
5. (Currently amended) The method of claim 1, wherein the indicator is stored in a performance instrumentation shadow cache and wherein the processor checks the ~~instructions performance~~ instrumentation shadow cache to determine whether the indicator is associated with the instructions.
6. (Original) The method of claim 1, wherein the instruction is received in a bundle by an instruction cache in the processor and wherein the indicator comprises at least one spare bit in a field in the bundle.
7. (Original) The method of claim 1, wherein the indicator is a separate instruction.

8. (Original) The method of claim 1, wherein an event in the events includes at least one of an entry into a module, an exit from a module, an entry into a subroutine, an exit from a subroutine, an entry into a function, an exit from a function, a start of input/output, and a completion of input/output, the execution of the instruction.
9. (Original) The method of claim 1, wherein the determining step comprises:  
determining, by an instruction cache, whether the indicator is present in a field within the instruction.
10. (Original) The method of claim 1, wherein the enabling step comprises:  
sending a signal to a performance monitor unit, wherein the performance monitor unit counts each event associated with execution of the instruction using a counter.
11. (Currently amended) A data processing system comprising:  
a performance monitor unit, wherein the performance monitor unit counts events for an instruction when a signal is received; and  
an instruction cache, wherein the instruction cache receives instructions and sends the signal to the performance monitor unit to count events associated with the instruction when an indicator is associated with the instruction ~~is associated with the indicator~~.
12. (Original) The data processing system of claim 11, wherein the instruction is located in a bundle received by the instruction cache.
13. (Original) The data processing system of claim 12, wherein the indicator is located in the bundle.
14. (Original) The data processing system of claim 11, wherein the indicator is located in a performance instrumentation shadow memory in association with the instruction.
15. (Original) A method in a data processing system for monitoring access to data, the method comprising:  
identifying a memory location associated with an indicator; and  
enabling counting of events associated with accesses to the memory location.

16. (Original) The method of claim 15, wherein the enabling step comprises:  
sending a signal from a data cache to a performance monitor unit to enable the performance monitor unit to count events associated with accesses to the memory location; and  
incrementing a counter in a performance memory unit each time an event associated with an access to the memory location occurs.
17. (Original) The method of claim 15, wherein an event in the events includes access to the memory location.
18. (Currently amended) A data processing system for processing instructions, the data processing system comprising:  
determining means, responsive to receiving an instruction at a processor in the data processing system, for determining whether an indicator is associated with the instruction; [[and]]  
enabling means for enabling counting of each event associated with execution of the instruction if the indicator is associated with the instruction; and  
counting means for counting each event associated with the execution of the instruction if counting is enabled for the instruction.
19. (Canceled)
20. (Currently amended) The data processing system of claim [[19,]] 18, wherein the counting means comprises:  
incrementing means for incrementing a counter associated with the indicator each time the event occurs.
21. (Original) The data processing system of claim 18, wherein the receiving means comprises:  
means for receiving the instruction in an instruction cache in the processor.
22. (Original) A data processing system for monitoring access to data, the data processing system comprising:  
identifying means for identifying a memory location associated with an indicator; and  
enabling means for enabling counting of events associated with accesses to the memory location.

23. (Currently amended) A computer program product in a computer readable medium for processing instructions, the computer program product comprising:

first instructions, responsive to receiving an instruction at a processor in the data processing system, for determining whether an indicator is associated with the instruction; [[and]]

second instructions for enabling counting of each event associated with execution of the instruction if the indicator is associated with the instruction; and

third instructions for counting each event associated with the execution of the instruction if counting is enabled for the instruction.

24. (Canceled)

25. (Currently amended) The computer program product of claim [[24,]] 23, wherein the third instruction comprises:

sub-instructions for incrementing a counter associated with the indicator each time the event occurs.